WHAT IS CLAIMED IS:

1.   A method for accelerating reception by a user browser of a target original object (OO) having an original object content, the OO transmitted from a originating server over a network in response to a request by the user, the method comprising steps of:

a.   using an accelerator to determine a fractional content of the target OO that is not cached in a local browser cache, said determination including comparisons with a plurality of same domain objects

b.   transmitting said fractional content to the user, and

c.   at the user browser, fulfilling the request by using said fractional content and additional content received by the browser from said accelerator and from the server.

2.   The method of claim 1, wherein said step of using an accelerator to determine a fractional content includes

      i.   building a static object repository (SOR) that includes static objects (SOs),

      ii.   finding a static object list (SOL) in said SOR that best fits the target OO, and

      iii.   building a dynamic object (DO) that includes reference to each static object in said SOL as well as additional information, said dynamic object characterized by having less content than said target OO content,

and wherein said step of transmitting includes transmitting said dynamic object to the user.

3.   The method of claim 2, wherein said building a SOR is preceded by preparing an original object repository (OOR), wherein said building a SOR is performed by analyzing and comparing all OOs in said OOR, and wherein said finding a SOL that best fits the target OO includes running a comparison between the target OO and each of the SOs in the SOL using a URL of the target OO as a hint.

4.   The method of claim 2, wherein said additional information in said DO includes:

     i.   a reference to a browser rendering object (BRO) having a BRO script that includes a substring offset, a length of the string to be extracted from the target OO and an offset, and

     ii.   a dynamic data array that consists of an unmatched areas between the target OO and any SO from said SOL, and pointers to a location inside each said SO that is matched within the target OO.

5.   The method of claim 4, wherein said dynamic data array further includes:

     A.   a refernce to each said SO in said SOL,

     B.   a substring offset (the starting point of a string to be extracted from said SO and inserted into the target OO);

     C.   the length of the string to be extracted from said SO; and

     D.   an offset (i.e. placement location) in the container page.

6.   The method of claim 1, wherein each said original object is a Web page.

7.   A method for accelerating transmission of objects between a source server having a domain and an end-user browser over a network, the method comprising steps of:

     a.   providing an accelerator that communicates with the server and the end-user,

     b.   receiving, by said accelerator, a target original object (OO) from the server in response to an end user request,

     c.   processing, by said accelerator, said target OO to produce a reduced content dynamic object (DO) that includes references to matching static objects (SOs) stored in a local cache of said browser, dynamic objects that need refreshing, and reassembly instructions for reassembling said target OO,

     d.   transmitting said DO to the end-user, and

     e.   fulfilling said request at the end user browser by reassembling said target OO using said DO references, objects and reassembly

instructions and additional target object components missing from said local cache.

8.   The method of claim 7, wherein said step of processing of said target OO includes identifying a static object list (SOL) that may fit said target OO, and wherein said step of transmitting includes transmitting references to SOs in said SOL instead of transmitting said SOs to the end-user, thereby reducing the amount of traffic between the source server and the end user.

9.   The method of claim 8, wherein said identifying of a static object list (SOL) that may fit said target OO is preceded by building, using previous OOs requested from the domain, an original object repository (OOR) in which each entry contains a pointer to OO attributes such as a request URL and a list of pointers to a SO score-of-match level, and wherein said identifying further includes building, preferably periodically, a static object repository (SOR) by analyzing and comparing all the objects in said OOR, said SOR used to form said SOL.

10.   A method for accelerating traffic over the Internet, comprising steps of:
    a.   positioning an accelerator between a source server and an end user having a browser with a browser cache, said accelerator operative to process requests from the end user and target original objects (OO) served by the server in response to user requests,
    b.   processing each said target OO to produce a dynamic object (DO),
    c.   transmitting said DO to the end user, and
    d.   reassembling at the end user the target OO using said DO and additional required data not stored in said cache.

11.   The method of claim 10, wherein said step of processing includes fragmenting each said target OO into dynamic and static components and determining which of said static components do not already exist in said browser cache and which of said dynamic components need refreshing, and wherein said step of transmitting includes transmitting with said DO only said static components or parts that do not already exist in said browser cache and said dynamic components that need refreshing.

12. The method of claim 11, wherein said determining is done heuristically, based on a plurality of OOs previously received by said accelerator from the same domain in the Internet.

13. The method of claim 11, wherein said fragmenting is done heuristically.

14. The method of claim 11, wherein said target OO is a Web page.

15. A method for accelerating traffic between a server and an end-user over a network, comprising steps of:

a. obtaining, at an accelerator interposed between the server and the end user, a request from the end-user for a target object and from the server said target object ,

b. fragmenting said target object into target dynamic and static components,

c. identifying, in an end-user cache, static components similar to said target static components,

d. transmitting to the end-user only non-similar target static components that do not have closely fitting matches in said end-user cache as well as updated dynamic components, and

e. reassembling, at the end-user, said target object using said transmitted static components and said updated dynamic components.

16. The method of claim 15, wherein said network is the Internet and wherein said target object is a Web page.

17. A system for accelerating traffic over a network between a server serving a response object in response to a request from an end-user comprising:

a. a mechanism for determining fresh static and dynamic components in a most recent of the response objects served by the server, said fresh components differing substantially from previous components sent to the end-user in previous response objects and stored in an end-user cache,

b.   a mechanism for transmitting said fresh components to the end-
user, and

c.   a mechanism for reassembling said most recent response object
at the end-user using said transmitted fresh components and required
components of said recent response object not cached in said cache.

18.   The system of claim 17, wherein said mechanism for determining fresh static
and dynamic components in a most recent of the response objects served by the server
includes an accelerator operative to fragment a requested object into static and
dynamic components and perform a comparison between said static components and
said end–user cached components to identify said fresh components, and wherein said
mechanism for transmitting said fresh components to the end-user includes a dynamic
object having including said fresh components, reference to fitting said previous
components and reassembly instructions.